

CLAIMS

We claim:

1. A system, comprising:

5

one or more processors;

memory coupled to the one or more processors and configured to store program
instructions executable by the one or more processors to implement:

10

one or more applications configured to initiate one or more transactions,
wherein each of the one or more transactions comprises requests to
access one or more data sources; and

15

a transaction manager configured to manage the one or more transactions
initiated by the one or more applications;

20

wherein for each transaction, the transaction manager is configured to
request permission to change the state of the transaction, and
wherein the transaction manager is configured to not change the
state of the transaction without said permission.

25

2. The system as recited in claim 1, further comprising a transaction freeze
manager configured to pause the transaction manager in response to a pause request by
withholding said permission and resume the transaction manager in response a resume
request by granting said permission.

3. The system as recited in claim 2, wherein the transaction freeze manager
is a part of the transaction manager.

30

4. The system as recited in claim 2, wherein the transaction freeze manager is configured to receive requests to pause the transaction manager from an administrative entity.

5 5. The system as recited in claim 2, wherein the transaction freeze manager is configured to queue received state transition permission requests and transaction manager pause requests in the order received.

10 6. The system as recited in claim 5, wherein the transaction freeze manager is configured to service queued state transition permission requests and transaction manager pause requests in FIFO order.

15 7. The system as recited in claim 2, wherein the transaction freeze manager is configured to grant the state transition permission request if the transaction manager is not paused.

8. The system as recited in claim 2, wherein the transaction freeze manager is configured to grant the pause request if the transaction manager is not paused and there are no outstanding state transition permission requests received prior to the pause request.

20

9. The system as recited in claim 2, wherein the transaction freeze manager is configured to not grant requests if the transaction manager is paused.

10. A system, comprising a plurality of computer systems coupled by one or 25 more networks, wherein the plurality of computer systems comprise:

one or more processors; and

30 memory coupled to the one or more processors and configured to store program instructions executable by the one or more processors to implement one or more application servers comprising:

one or more applications configured to initiate one or more transactions,
wherein each of the one or more transactions comprises requests to
access one or more data sources; and

5

one or more transaction managers configured to manage the one or more
transactions initiated by the one or more applications;

10 wherein for each transaction, the one or more transaction managers are
configured to request permission to change the state of the
transaction, and wherein the one or more transaction managers are
configured to not change the state of the transaction without said
permission.

15 11. A system, comprising:

one or more processors;

20 memory coupled to the one or more processors and configured to store program
instructions executable by the one or more processors to implement:

one or more applications configured to initiate one or more transactions,
wherein each of the one or more transactions comprises requests to
access one or more data sources; and

25

a transaction manager configured to manage the one or more transactions
initiated by the one or more applications;

30 wherein for each transaction, the transaction manager is configured to
request a read lock on a transaction freeze object to change the
state of the transaction, and wherein the transaction manager is

configured to not change the state of the transaction without said lock.

12. The system as recited in claim 11, further comprising a transaction freeze
5 manager configured to pause the transaction manager in response to a pause request by withholding read locks and resume the transaction manager in response a resume request by granting read locks.

13. The system as recited in claim 12, wherein the transaction freeze manager
10 is a part of the transaction manager.

14. The system as recited in claim 12, wherein the transaction freeze manager is configured to receive requests for write locks on the transaction freeze object from an administrative entity to pause the transaction manager.
15

15. The system as recited in claim 12, wherein the transaction freeze manager is configured to queue received lock requests in the order received.

16. The system as recited in claim 15, wherein the transaction freeze manager
20 is configured to service queued lock requests in FIFO order.

17. The system as recited in claim 12, wherein the transaction freeze manager is configured to grant read locks if the transaction manager is not paused.

25 18. The system as recited in claim 12, wherein the transaction freeze manager is configured to grant a write lock if the transaction manager is not paused and there are no outstanding read lock requests received prior to the write lock request.

19. The system as recited in claim 12, wherein the transaction freeze manager
30 is configured to not grant locks if a write lock on the transaction freeze object is currently held by an administrative entity.

20. A system, comprising a plurality of computer systems coupled by one or more networks, wherein the plurality of computer systems comprise:

5 one or more processors; and

memory coupled to the one or more processors and configured to store program instructions executable by the one or more processors to implement one or more application servers comprising:

10 one or more applications configured to initiate one or more transactions, wherein each of the one or more transactions comprises requests to access one or more data sources; and

15 one or more transaction managers configured to manage the one or more transactions initiated by the one or more applications;

20 wherein for each transaction, the one or more transaction managers are configured to request a read lock on a transaction freeze object to change the state of the transaction, and wherein the one or more transaction managers are configured to not change the state of the transaction without said lock.

21. A method, comprising:

25 receiving a pause request;

pausing a transaction manager in response to the pause request by withholding permission to change the state of one or more transactions managed by the
30 transaction manager.

receiving a plurality of resume requests; and

resuming the transaction manager in response the resume request by granting permission to change the state of the one or more transactions managed by
5 the transaction manager.

22. The method as recited in claim 21, wherein a transaction freeze manager grants and withholds said permission.

10 23. The method as recited in claim 22, wherein the transaction freeze manager is a part of the transaction manager.

15 24. The method as recited in claim 22, wherein the transaction freeze manager is configured to receive requests to pause the transaction manager from an administrative entity.

25. The method as recited in claim 22, wherein the transaction freeze manager is configured to queue received state transition permission requests and transaction manager pause requests in the order received.

20 26. The method as recited in claim 25, wherein the transaction freeze manager is configured to service queued state transition permission requests and transaction manager pause requests in FIFO order.

25 27. The method as recited in claim 22, wherein the transaction freeze manager is configured to grant a state transition permission request if the transaction manager is not paused.

30 28. The method as recited in claim 22, wherein the transaction freeze manager is configured to grant a transaction manager pause request if the transaction manager is

not paused and there are no outstanding state transition permission requests received prior to the pause request.

29. The method as recited in claim 22, wherein the transaction freeze manager
5 is configured to not grant requests if the transaction manager is paused.

30. A method, comprising:

receiving a pause request;

10

pausing a transaction manager in response to the pause request by withholding
read locks on a transaction freeze object.

receiving a resume request; and

15

resuming the transaction manager in response to the resume request by granting read
locks on the transaction freeze object.

31. The method as recited in claim 30, wherein a transaction freeze manager
20 grants and withholds the read locks.

32. The method as recited in claim 31, wherein the transaction freeze manager
is a part of the transaction manager.

25 33. The method as recited in claim 31, wherein the transaction freeze manager
is configured to receive requests for write locks on the transaction freeze object to pause
the transaction manager from an administrative entity.

34. The method as recited in claim 31, wherein the transaction freeze manager
30 is configured to queue received lock requests in the order received.

35. The method as recited in claim 34, wherein the transaction freeze manager is configured to service queued lock requests in FIFO order.

36. The method as recited in claim 31, wherein the transaction freeze manager
5 is configured to grant a read lock if the transaction manager is not paused.

37. The method as recited in claim 31, wherein the transaction freeze manager is configured to grant a write lock if the transaction manager is not paused, and there are no outstanding read lock requests received prior to the write lock request, and there are
10 no outstanding read locks.

38. The method as recited in claim 31, wherein the transaction freeze manager is configured to not grant locks if a write lock on the transaction freeze object is currently held by an administrative entity.

15

39. A carrier medium comprising program instructions, wherein the program instructions are computer-executable to:

receive a pause request;

20

pause a transaction manager in response to the pause request by withholding permission to change the state of one or more transactions managed by the transaction manager.

25

receive a resume request; and

resume the transaction manager in response to the resume request by granting permission to change the state of the one or more transactions managed by the transaction manager.

30

40. The carrier medium as recited in claim 39, wherein a transaction freeze manager grants and withholds said permission.

41. The carrier medium as recited in claim 40, wherein the transaction freeze manager is a part of the transaction manager.

42. The carrier medium as recited in claim 40, wherein the transaction freeze manager is configured to receive requests to pause the transaction manager from an administrative entity.

10

43. The carrier medium as recited in claim 40, wherein the transaction freeze manager is configured to queue received state transition permission requests and transaction manager pause requests in the order received.

15

44. The carrier medium as recited in claim 43, wherein the transaction freeze manager is configured to service queued state transition permission requests and transaction manager pause requests in FIFO order.

20

45. The carrier medium as recited in claim 40, wherein the transaction freeze manager is configured to grant a state transition permission request if the transaction manager is not paused.

25

46. The carrier medium as recited in claim 40, wherein the transaction freeze manager is configured to grant a transaction manager pause request if the transaction manager is not paused and there are no outstanding state transition permission requests received prior to the pause request.

47. The carrier medium as recited in claim 40, wherein the transaction freeze manager is configured to not grant requests if the transaction manager is paused.

30

48. A carrier medium comprising program instructions, wherein the program instructions are computer-executable to:

receive a pause request;

5

pause a transaction manager in response to the pause request by withholding read locks on a transaction freeze object.

receive a resume request; and

10

resume the transaction manager in response to the resume request by granting read locks on the transaction freeze object.

49. The carrier medium as recited in claim 48, wherein a transaction freeze manager grants and withholds the read locks.

50. The carrier medium as recited in claim 49, wherein the transaction freeze manager is a part of the transaction manager.

20 51. The carrier medium as recited in claim 49, wherein the transaction freeze manager is configured to receive requests for write locks on the transaction freeze object to pause the transaction manager from an administrative entity.

25 52. The carrier medium as recited in claim 49, wherein the transaction freeze manager is configured to queue received lock requests in the order received.

53. The carrier medium as recited in claim 52, wherein the transaction freeze manager is configured to service queued lock requests in FIFO order.

30 54. The carrier medium as recited in claim 49, wherein the transaction freeze manager is configured to grant a read lock if the transaction manager is not paused.

55. The carrier medium as recited in claim 49, wherein the transaction freeze manager is configured to grant a write lock if the transaction manager is not paused, and there are no outstanding read lock requests received prior to the write lock request, and
5 there are no outstanding read locks.

56. The carrier medium as recited in claim 49, wherein the transaction freeze manager is configured to not grant locks if a write lock on the transaction freeze object is currently held by an administrative entity.